For Mr Olivier CARTON

Your student / Abdelrahman ELGAMAL

Java and C#

British Computer Society

Financial Services Specialist Group, 23rd April 2002

Kingston and Croydon Branch, 15th October 2002

Brian Shearing

Shape of talk

- Origins of Java
- Origins of C#
- Java and C# compared
- Java and C# considered

Part I

Origins of Java

Early history of Java

1992

project: Green
product: Star *7
language: Oak

"to identify convergence of digitally controlled consumer devices and computers"

1993

Mosaic

1994

Webrunner (HotJava)

hoped for 10,000 downloads; response killed Sun net server; 2,000 email requests/day

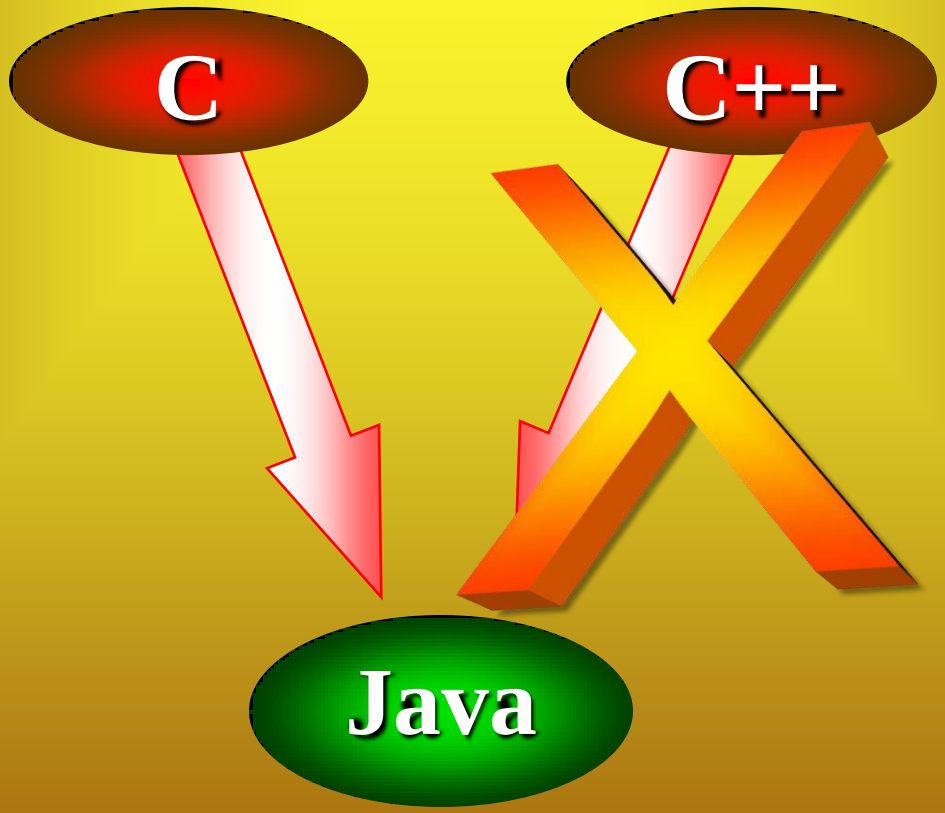
1995

α Java posted on net

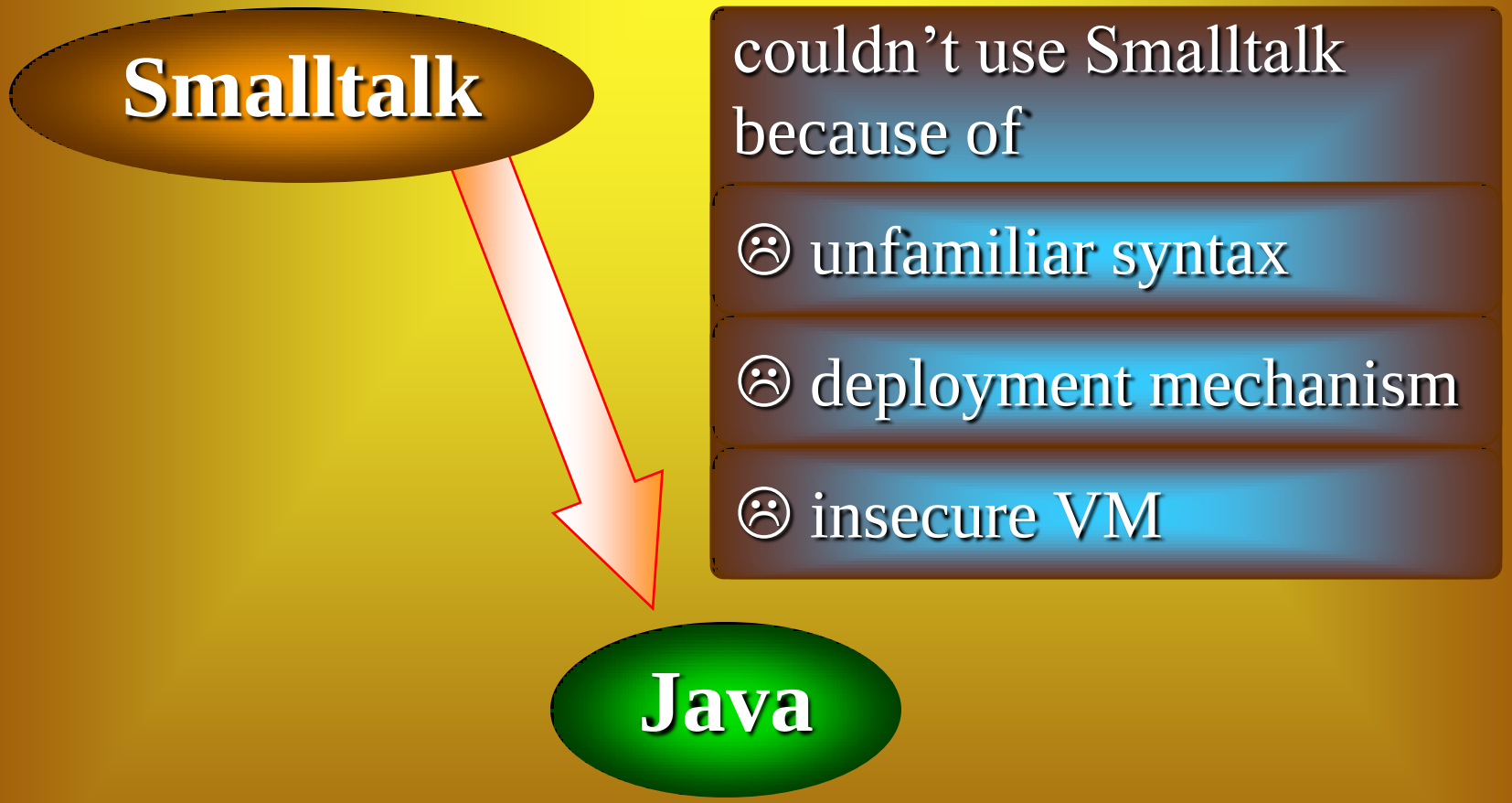
James Gosling

Sun

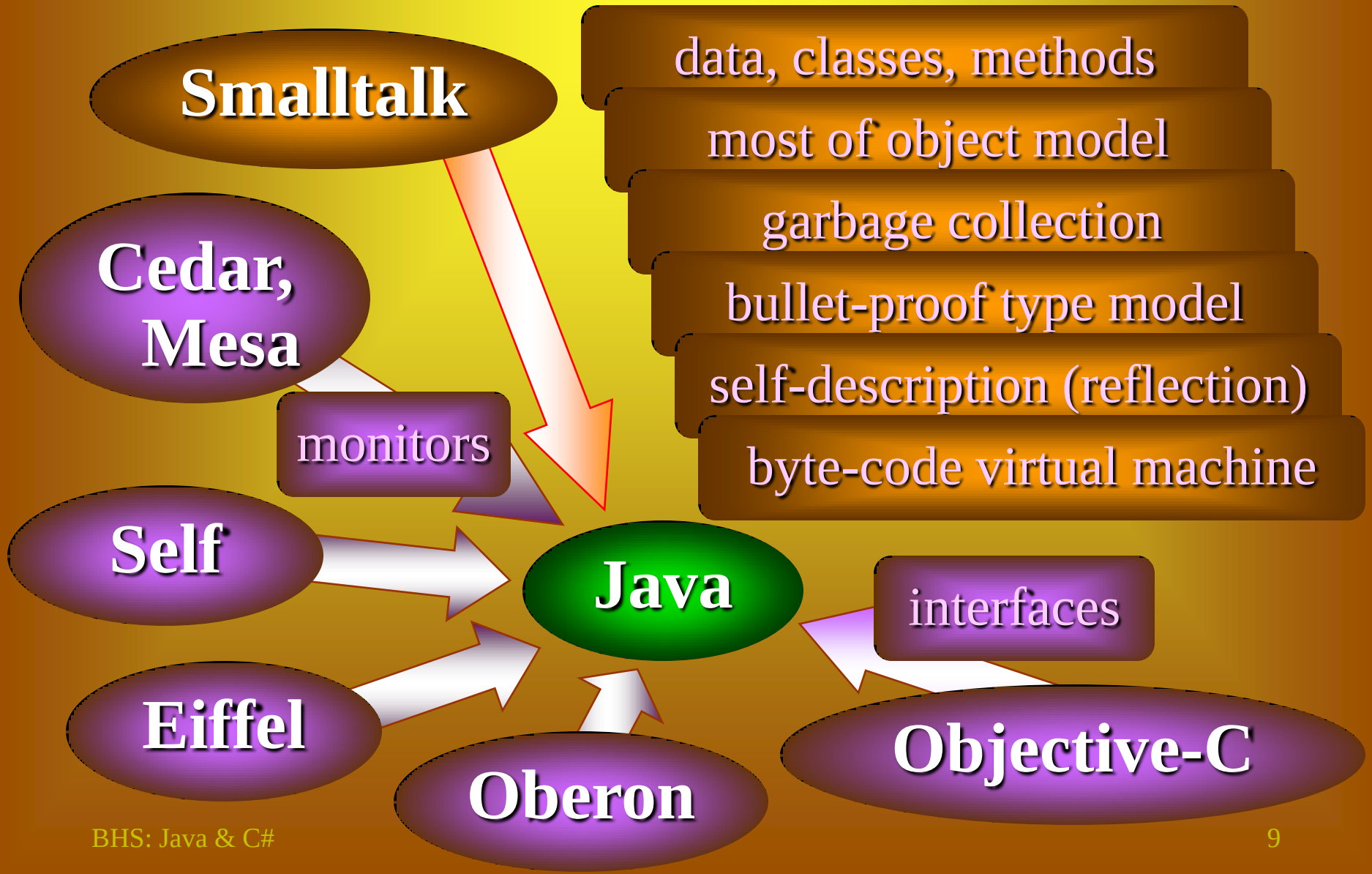
Influences on Java



Influences on Java



Influences on Java



Part II

Origins of C#

Anders Hejlsberg
Microsoft

Design goals of C#

C# is not a Java clone.

In the design of C#, we looked at a lot of languages.

We looked at C++, we looked at Java, at Modula2, C, and we looked at Smalltalk.

Design goals of C#

One of the key differences between C# and these other languages, particularly Java, is that we tried to stay much closer to C++ in our design.

C# borrows most of its operators, keywords, and statements directly from C++.

Part III

Java and C# compared

Design goals of Java

familiarity

C syntax

curly braces:
`{ }`

illogical semicolon:
`if (…) …; else …;`

dangling *else*:
`if (…) if (…) …; else …;`

adopted for C#

Example of Java

```
class Hello {
   public static void main(String[] params) {
      System.out.println("Hi " + params[0]);
   }
}
```

```
C:\TSF\Play>javac Hello.java
```

```
C:\TSF\Play>java Hello Brian
Hi Brian
```

Example in Java and C#

```
class Hello {
   public static void main(String[] params) {
      System.out.println("Hi " + params[0]);
   }
}
```

```
class Hello {
   public static void Main(string[] params) {
      System.Console.WriteLine("Hi " + params[0]);
   }
}
```

Example of C#

```
C:\TSF\Play>csc Hello.cs
```

```
C:\TSF\Play>Hello Betty
Hi Betty
```

```
class Hello {
   public static void Main(string[] params) {
      System.Console.WriteLine("Hi " + params[0]);
   }
}
```

Result of compiling Java and C#

compiled Java byte-codes
>javac Hello.java

```
C:\TSF\Play>dir

21/04/2002  12:54               127 Hello.java
21/04/2002  12:30               133 Hello.cs
21/04/2002  16:58               569 Hello.class
21/04/2002  17:17             3,072 Hello.exe
```

compiled .net program
>csc Hello.cs

Dis-assembled Java

```
C:\TSF\Play>javap Hello
```

Class ‘Hello’ has a default constructor and method ‘main’

```
Compiled from Hello.java
class Hello extends java.lang.Object {
    Hello();
    public static void main(java.lang.String[]);
}
```

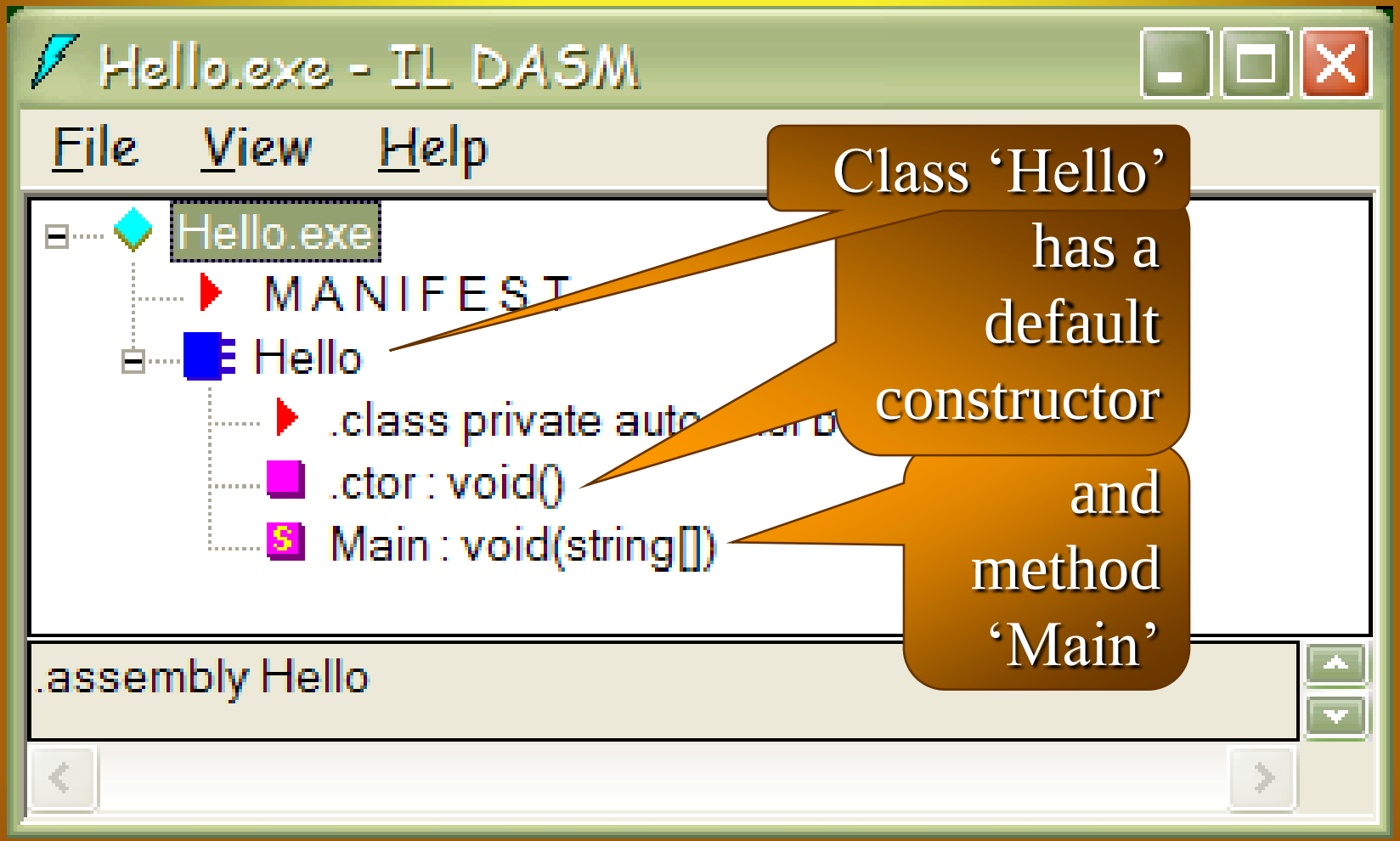
Dis-assembled Java

```
Method void main(java.lang.String[])
  0 getstatic #2 <Field java.io.PrintStream out>
  3 new #3 <Class java.lang.StringBuffer>
  6 dup
  7 invokespecial #4 <Method java.lang.StringBuffer()>
 10 ldc #5 <String "Hi ">
 12 invokevirtual #6
      <Method java.lang.StringBuffer append(java.lang.String)>
 15 aload_0
 16 iconst_0
 17 aaload
 18 invokevirtual #6
      <Method java.lang.StringBuffer append(java.lang.String)>
 21 invokevirtual #7 <Method java.lang.String toString()>
 24 invokevirtual #8 <Method void println(java.lang.String)>
 27 return
```

```
C:\TSF\Play>javap Hello -c
```

Dis-assembled C#

```
C:\TSF\Play>ildasm Hello.exe
```



Dis-assembled C#


```
.method public hidebysig static void
                        Main(string[] parameters) cil managed
{
  .entrypoint
  // Code size       19 (0x13)
  .maxstack  8
  IL_0000:  ldstr      "Hi "
  IL_0005:  ldarg.0
  IL_0006:  ldc.i4.0
  IL_0007:  ldelem.ref
  IL_0008:
      call string [mscorlib]System.String::Concat(string, string)
  IL_000d:
      call void [mscorlib]System.Console::WriteLine(string)
  IL_0012:  ret
} // end of method Hello::Main
```

Design goals of Java

architecture neutral

Java source

Java compiler

byte codes

class loader
(platform-specific JIT
code-generator)

Design goals of Java

adopted for C#

architecture neutral

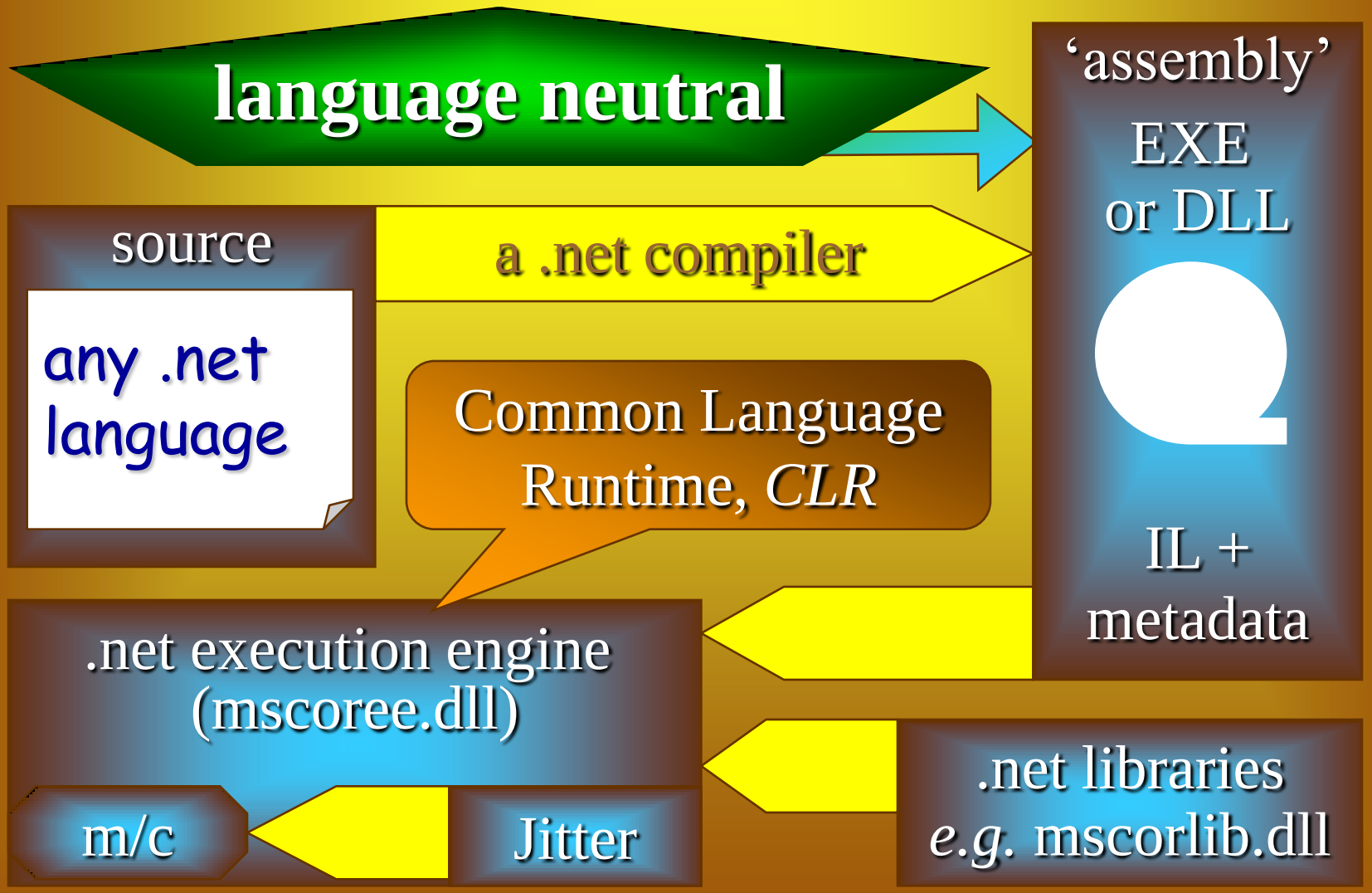
C# source

C# compiler

byte codes

class loader (platform-specific JIT code-generator)

Design goals of .NET



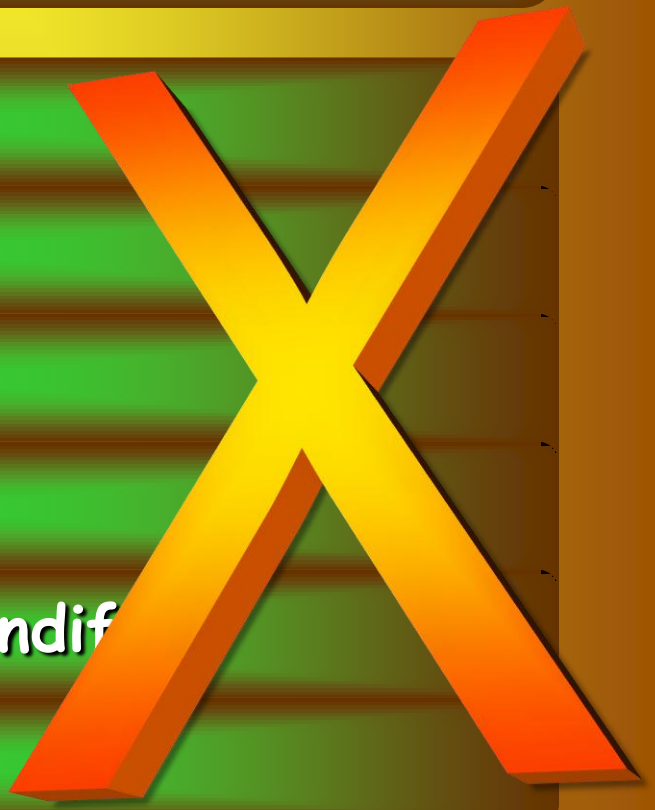
Design goal of Java—Simplicity

only one way to do most things

no **struct**, no **enum**, only **class**

no more write-only software

- **typedef**
- pre-processor
- **#define #undef**
- header files
- **#if, #elif, #else, #endif**
- operator overloading



Design goal of C#—Convenience

if two ways help, provide them

struct *and* enum *and* class

typedef ✗

pre-processor ✗

#define #undef ☺

header files ✗

#if, #elif, #else, #endif ☺

operator overloading ☺

also, conditional methods ...

Design goal of C#—Convenience

Square bracket syntax provides "Attributed programming"

—a form of *Aspect-oriented Programming*

```
[Conditional("Debug")]
public static void Assert(bool cond, String s) {
    if (!cond) {
        throw new AssertionException(s);
    }
}
```

C#—Attributes

```
[XmlRoot("Order", Namespace="urn:acme.b2b-schema.v1")]
public class PurchaseOrder
{
   [XmlElement("shipTo")]      public Address ShipTo;
   [XmlElement("billTo")]      public Address BillTo;
   [XmlElement("comment")]     public string Comment;
   [XmlElement("items")]       public Item[] Items;
   [XmlAttribute("date")]      public DateTime OrderDate;
}
```

C#—Events

```
public delegate void EventHandler(object sender, EventArgs e);

public class Button {
   public event EventHandler Click;

   protected void OnClick(EventArgs e) {
      if (Click != null) Click(this, e);
   }
}

public class MyForm: Form {
   Button okButton;

   public MyForm() {
      okButton = new Button(...);
      okButton.Caption = "OK";
      okButton.Click += new EventHandler(OkButtonClick);
   }

   void OkButtonClick(object sender, EventArgs e) {
      ShowMessage("You pressed the OK button");
   }
}
```

C#—Properties

```
public class Button: Control {

    private string caption;

    public string Caption {
        get {
            return caption;
        }
        set {
            caption = value;
            Repaint();
        }
    }
}
```

```
Button b = new Button();
b.Caption = "OK";
String s = b.Caption;
```

Design goals of C#

Sure, we can express these concepts by methods.

It's just harder, and there's more housekeeping.

We just think the time is right for a language that makes it easier to create components.

Design goal of C#—Convenience

Microsoft's tools are very good at doing the simple applications.

You can do the thing that follows their paradigm really quickly.

But as soon as you try to scale up, you get into trouble.

Types

Java

- value types
  - primitives (**boolean**, **int** …)
- reference types
  - classes, interfaces, arrays

C#

- value types
  - primitives (**boolean**, **int**, **uint** …)
  - enums (**enum Colour { red, green }**)
  - structs (**struct Pair { double u, v }**)
- reference types
  - classes, interfaces, arrays
  - delegates (**delegate void NonEmpty()**)

Design goals of Java

garbage collected

object-oriented

data only on objects and classes; no globals

methods only on objects and classes; no functions

methods polymorphic by default; unless **final**

adopted for C#

except

methods *not* polymorphic by default; use **virtual**, and then **override**

primitives are objects (automatic *boxing* and *unboxing*)

Design goals of Java

Java source

secure

byte codes

Java compiler

class loader
(platform-specific JIT
code-generator)

loader rechecks
correctness of
class

Java IL is
strongly
typed

Design goals of C#

Our IL is type-neutral.

There's no information in the instructions that specifies the type of the arguments.

Rather, that is inferred by what's been pushed on the stack.

This approach makes the IL more compact

...which in turn makes it easier to translate IL into native code.

C#

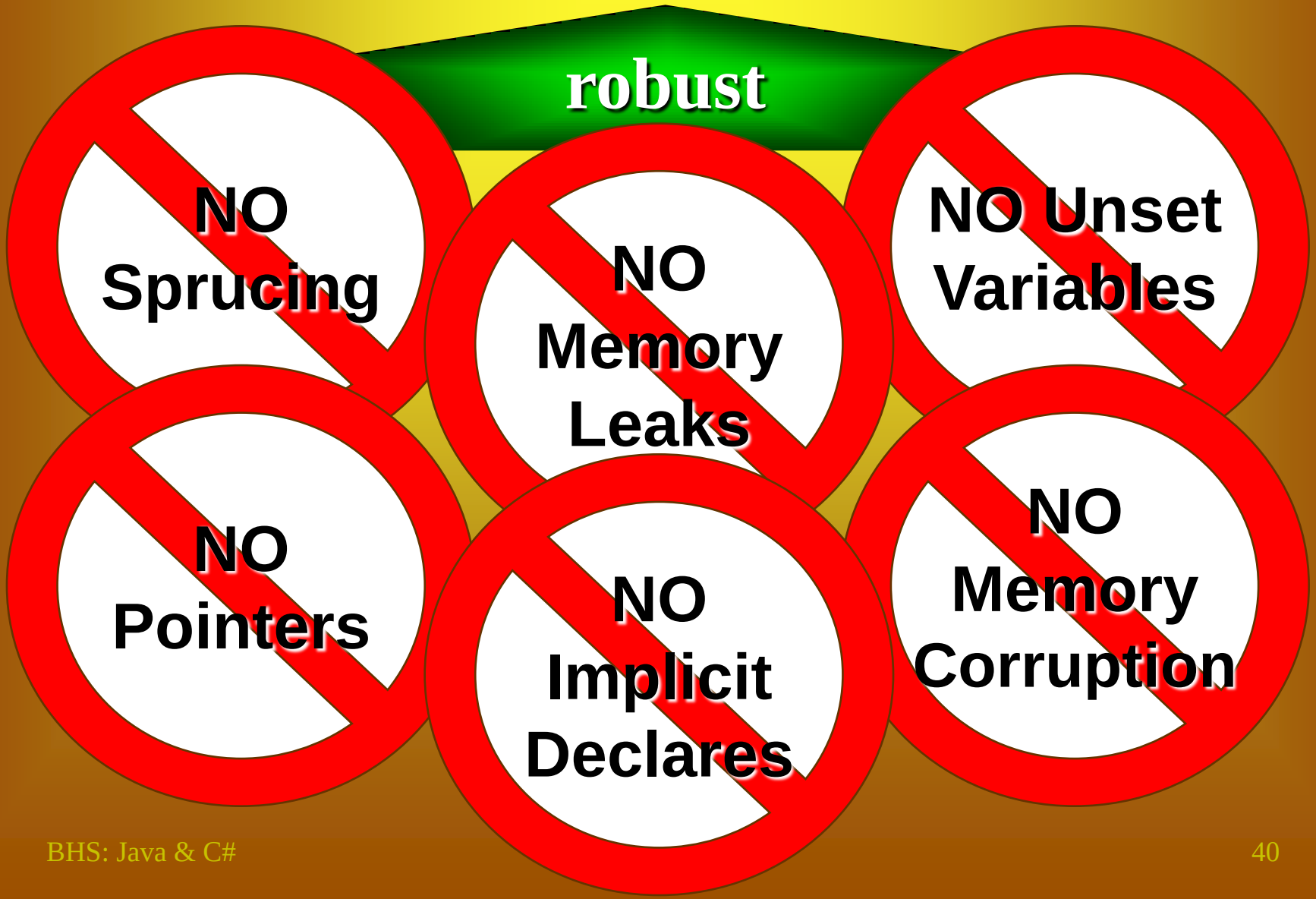
When the Internet exploded, Microsoft seemed ill-prepared to retrofit adequate security.

Chairman Bill Gates has issued a directive that, at long last, security should be more important than getting the next release out the door.

Microsoft originally built its operating system and applications for … the isolated office environment, where all the computers are assumed to belong to friendly colleagues, not adversaries.

I think they'll find they have a long road ahead of them.

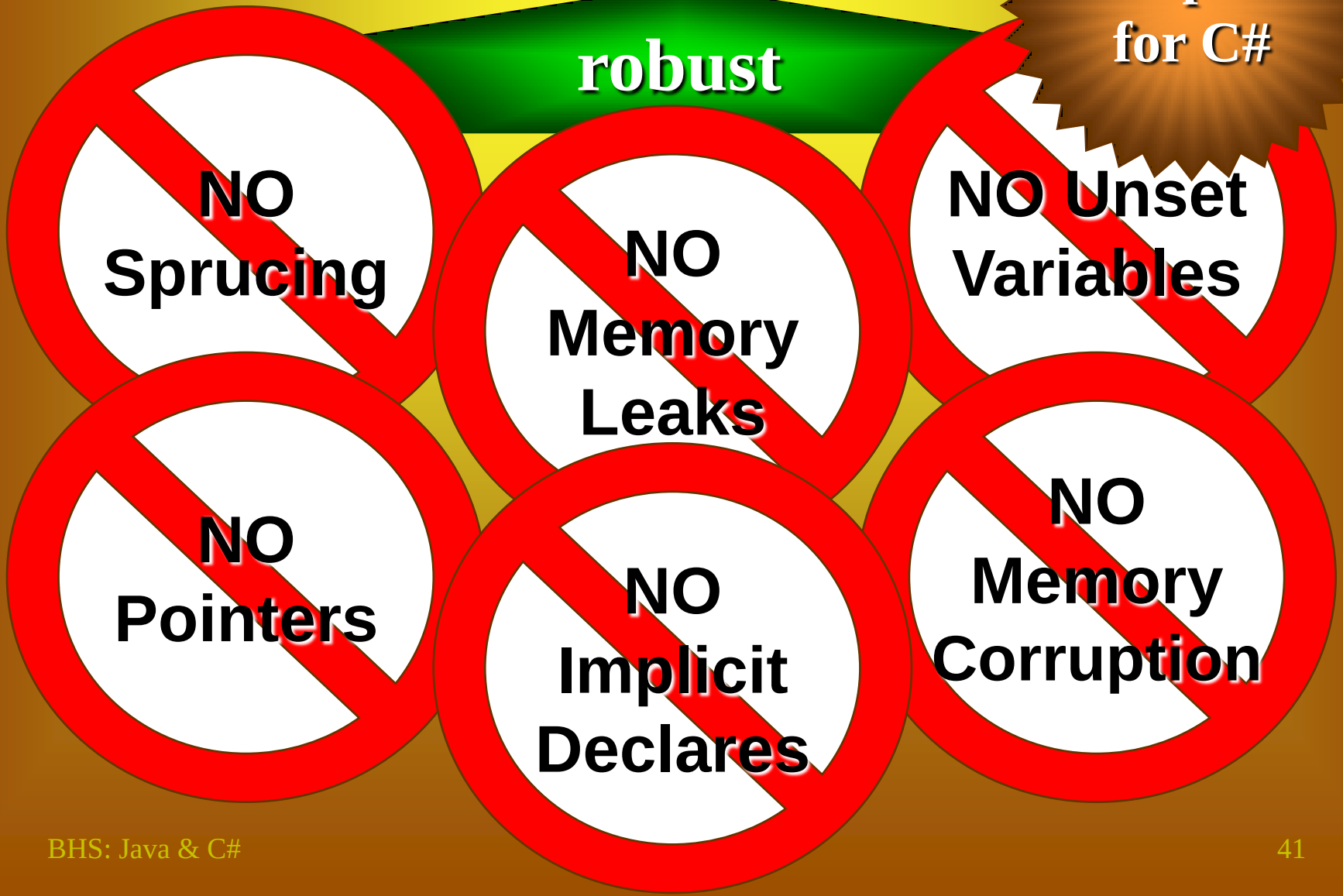
Design goals of Java



Design goals of Java

adopted for C#

robust



C#

unsafe

```
unsafe void LetsLiveDangerously() {
   char* buf = stackalloc char[256];
   for (char* p = buf; p < buf + 256; p++) *p = 0;
   ...
}
```

C#

"Unsafe code is in fact a 'safe' feature," the C# specification continues, "from the perspective of both developers and users. … the execution engine works to ensure that unsafe code cannot be executed in an untrusted environment."

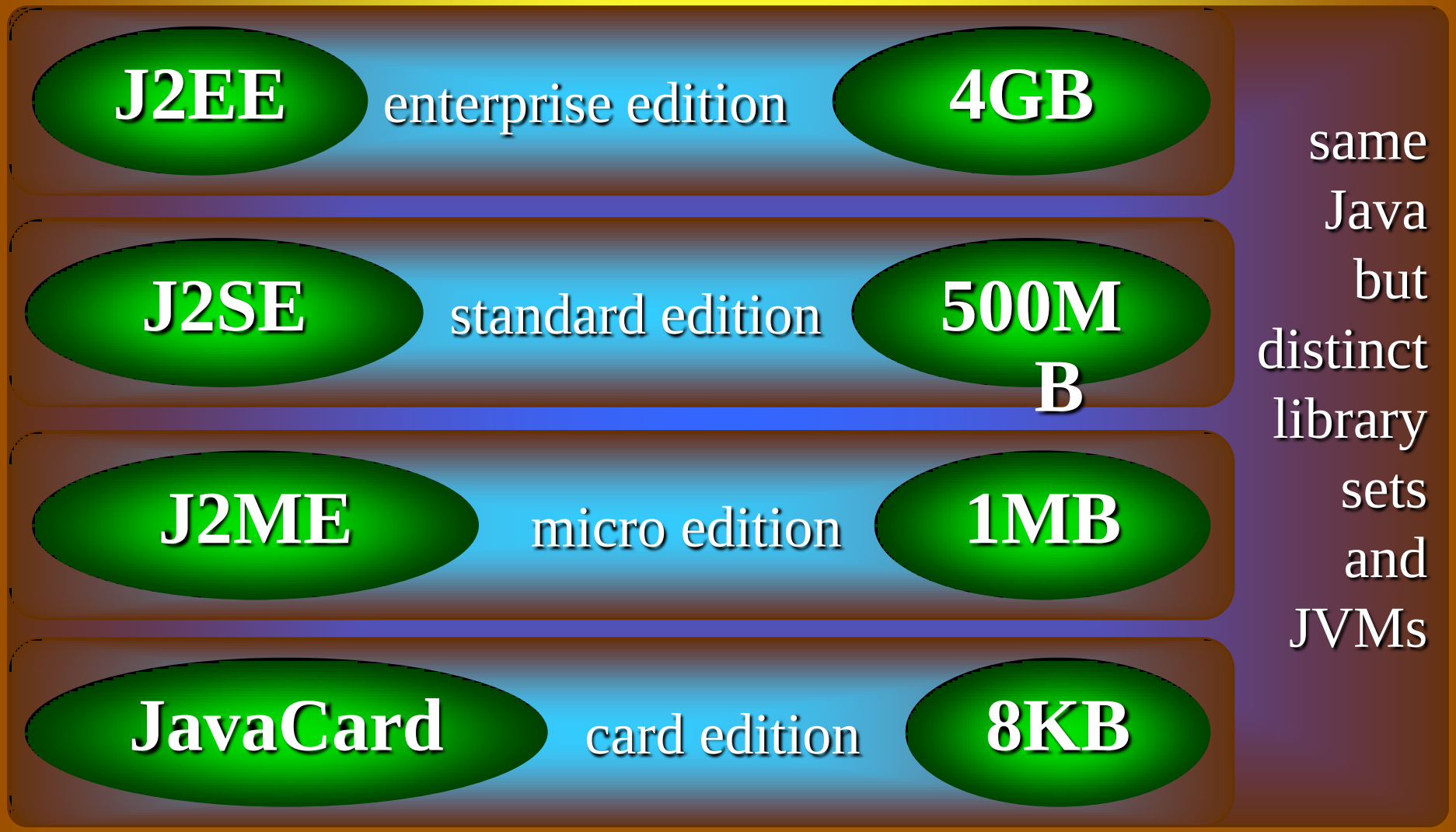
Did they get their design right this time?

I, for one, would bet against it.

C# is already cast in stone as an ECMA standard. And only now has Microsoft decided to make security a priority.

Adding security to an existing, large insecure system will, in my judgement, prove an impossible task.

Java editions



.NET editions

adopted for C#

.NET	enterprise edition	4GB
.NET	standard edition	500MB
.NET Compact	compact edition	1MB

same C# but distinct library sets and CLRs

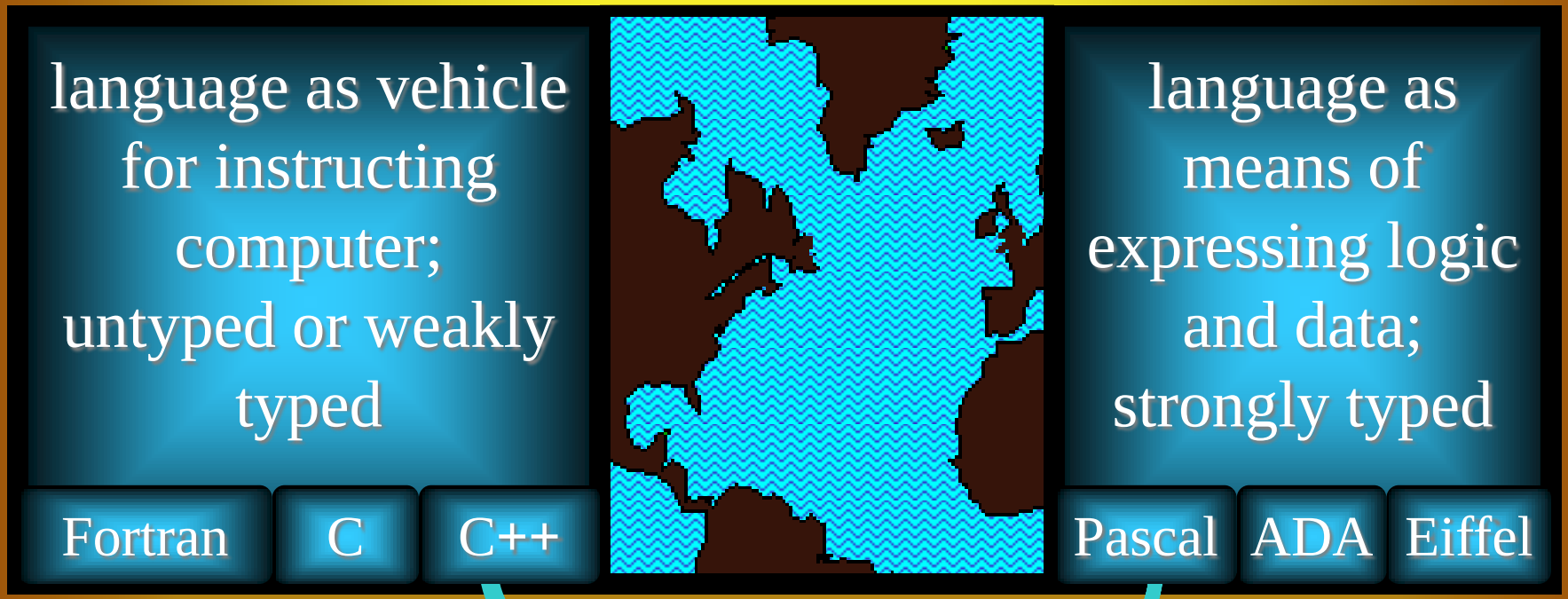
EcnoJIT

Part IV

Java and C# considered

Language traditions

less abstract — more abstract



C#

Our approach with C# has simply been to offer an alternative to C++ programmers who find that language too complicated.

and to Java programmers who miss certain features of C and C++ that were lost in the translation.

Java

We've historically worked on making the hard things possible and not worried so much on the easy things.

A big piece of the difficulty in designing it was to make it as simple as possible.

C#

The work that we've done with *attributes*—a feature used to add typed, extensible metadata to any object—is completely new and innovative.

I haven't seen it in any other programming language.

Java

Java is fast becoming an educational programming language.

If you go round to universities, high schools and middle schools, more often than not, they are teaching Java

C#

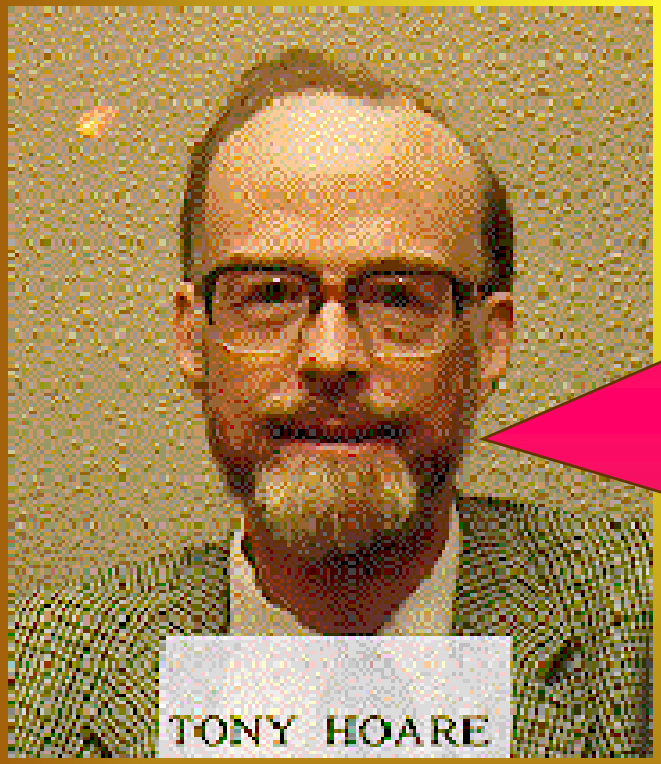
Who's helping COBOL programmers today?

Who's taking them to the Web?

Only on the .NET platform can you embed Fujitsu COBOL in an ASP page.

I mean it's truly revolutionary.

Java and C#



I'm also encouraged by recent languages like Java, which take program correctness as a specific goal—sometimes.

Should you use Java or C# ?

Both languages are strongly typed

object-oriented

garbage-collected

network savvy

component building

application development languages

and a pleasure to write.

Now there is no imaginable reason for using C or C++ for application development

(Not that there ever was.)

Java and C#

British Computer Society

Financial Services Specialist Group, 23rd April 2002

Kingston and Croydon Branch, 15th October 2002

Brian Shearing

ShearingBH@aol.com

Advanced Programming Specialist Group

A Tribute to E W Dijkstra

18.00 Thursday 14th November
Sun Microsystems
Regis House, London Bridge

Register: f_martin@lgu.ac.uk